

5

CLAIMS:

1. A method for specifying a system behavior, comprising:

- 10 (i) providing a system Graphic User Interface (GUI); the system GUI includes objects; the system GUI is associated with data structure describing at least the GUI objects; the method includes performing steps that include the following steps (ii) and (iii), as many times as required:
- 15 (ii) playing-in a scenario utilizing the system GUI, the scenario is representative of at least one use case; and
- (iii) constructing formal system behavior specification that corresponds to the scenario.

20 2. The method according to Claim 1, wherein said step (ii) further includes: operating at least one of said objects in the system GUI.

3. The method according to Claim 2, wherein said step further includes:

- (i) specifying user action by operating at least one of said objects;
- (ii) specifying environment action by operating at least one of said objects; and
- 25 (iii) specifying system reaction by operating at least one of said objects.

4. The method according to Claim 2, wherein said objects include at least one internal object and wherein said step further includes operating at least one internal object.

30 5. The method according to Claim 3, wherein said objects include at least one internal object and wherein said step further includes operating at least one internal object.

6. The method according to Claim 2, further comprising the step of: defining at least one control construct and wherein said step (iii) includes constructing formal system behavior specification that corresponds to the control construct.

- 5 7. The method according to Claim 6, wherein said control construct step includes creating generalization and loops selected from the group that includes dynamic loops, unbound loops and fixed loops.
8. The method according to Claim 1, wherein said step (iii) includes constructing symbolic messages.
- 10 9. The method according to Claim 1, further comprising the step of: reflecting in the system GUI the result of the played-in scenario.
10. The method according to Claim 1, wherein said formal system behavior specification at least one Live sequence chart (LSC).
11. The method according to Claim 1, wherein said formal system behavior specification being Temporal logic language.
- 15 12. The method according to Claim 1, wherein said formal system behavior specification being at least one Symbolic timing diagram.
13. The method according to Claim 1, wherein said formal system behavior specification being at least one Timed Buchi Automata.
- 20 14. The method according to Claim 1, further comprising, performing the following step as many times as required:
 - (iv) playing-out a scenario utilizing the system GUI and the system behavior specification.
- 25 15. The method according to Claim 2, further comprising the step of: defining at least one condition that may or must hold regarding the system, and wherein said step (c) includes constructing formal system behavior specification that corresponds to the at least one condition.
16. The method according to Claim 15, wherein at least one of said conditions includes defining condition regarding one or more of the operated objects.
- 30 17. The method according to Claim 10, further comprising the step of: selectively modifying at least one of said charts.
18. The method according to Claim 2, further comprising the step of: reflecting in the system GUI the result of the operation of at least one of said

5 objects.

19. An apparatus for specifying a system behavior, comprising:

a system Graphic User Interface (GUI) that includes objects and associated with data structure describing at least the GUI objects;

the apparatus includes a Play-Engine configured to perform at least the
10 following, as many times as required:

(i) playing-in a scenario utilizing the system GUI, the scenario is representative of at least one use case; and

(ii) constructing formal system behavior specification that corresponds to the scenario.

15 20. A method for playing out scenarios in a system, comprising:

(iii) providing a system Graphic User Interface (GUI) that includes objects; the system GUI is associated with data structure describing at least the GUI objects;

(iv) providing a formal system behavior specification;

20 performing the following step as many times as required:

playing-out a scenario utilizing the system GUI and the system behavior specification.

21. The method according to Claim 20, wherein said step (iii) includes: reflecting in the system GUI the result of the operation of said played-out

25 scenario.

5 22. The method according to Claim 20, wherein said formal system behavior specification being at least one Live sequence chart.

23. The method according to Claim 20, wherein said formal system behavior specification being Temporal logic language.

10 24. The method according to Claim 20, wherein said formal system behavior specification being at least one Symbolic timing diagram.

25. The method according to Claim 20, wherein said formal system behavior specification being at least one Timed Buchi Automata.

15 26. The method according to Claim 20, wherein said playing out includes: testing the behavior of the system directly from the system behavior specification.

27. The method according to Claim 25, wherein said testing includes running scenarios and forbidden scenarios.

28. The method according to Claim 20, further comprising the step of recording at least one played out scenario, constituting a run.

20 29. The method according to Claim 22, wherein said system behavior specification includes existential charts and universal charts, and wherein said universal charts include user action part and system reaction part.

25 30. The method according to Claim 20, further including the step of: indicating if the system behavior specification or portion thereof is successful or violated.

31. The method according to Claim 20, further including the step of providing a run that includes either or both of user and environment part and system reaction part, constituting a played scenario, and re-playing the run.

30 32. The method according to Claim 20, wherein said system behavior specification includes existential charts and universal charts, and wherein said universal charts include user action part, environment action part and system reaction part, and further comprising the step of, tracing either or both of said existential and universal charts, and indicating if a chart is successful or violated.

33. The method according to Claim 31, further comprising the step of

5 providing either or both of the user action part and environment action part of said run, replaying the run and indicating if the existential charts are successful or violated. ,

34. The method according to Claim 22, wherein said LSC charts include at least two live copies of the same chart simultaneously.

10 35. The method according to Claim 20, wherein said objects include at least one internal object and wherein said step further includes operating at least one internal object.

15 36. The method according to Claim 35, wherein said system GUI includes an object map and further comprising the step of: reflecting in the object map the result of the playing-out.

37. The method according to Claim 4, wherein said system GUI includes an object map and further comprising the step of: reflecting in the object map the result of the playing-in.

20 38. The method according to Claim 5, wherein said system GUI includes an object map and further comprising the step of: reflecting in the object map the result of the playing-in.

39. An apparatus for playing out scenarios in a system, comprising:
a system Graphic User Interface that includes objects and associated with data structure describing at least the GUI objects;
25 a formal system behavior specification;
a play-out engine configured to perform at least the following as many times as required: playing-out a scenario utilizing the system GUI and the system behavior specification.

30 40. The method according to Claim 14, wherein said playing-out is used to construct a prototype.

41. The apparatus according to Claim 39, wherein said playing-in is used to construct a prototype.

42. The method according to Claim 20, wherein said playing-out is used to construct a prototype.

5 43. The method according to Claim 14, wherein said playing-out is used to construct a tutorial.

44. The apparatus according to Claim 39, wherein said playing-out is used to construct a tutorial.

10 45. The method according to Claim 20, wherein said playing-out is used to construct a tutorial.

46. The method according to Claim 14, wherein said playing-out is used to construct a final implementation of a system.

47. The method according to Claim 20, wherein said playing-out is used to construct a final implementation of a system.

15 48. The apparatus according to Claim 30, wherein said playing-out is used to construct a final implementation of a system.

49. A computer program product that includes a storage medium storing a computer code for implementing the method steps of Claim 1.

20 50. A computer program product that includes a storage medium storing a computer code for implementing the method steps of Claim 20.

51. The method according to Claim 20, wherein said step (iii), includes animating interaction between GUI objects.

10034133-12204
F042321 ECH207